#include <bits/stdc++.h>

using namespace std;

int arr[4][5] = {{1, 4, 7, 10, 11},

{12, 15, 17, 21, 27},

{28, 31, 39, 43, 44},

{46, 48, 51, 55, 56}};

bool binary\_search2d(int start, int end, int rows, int col, int element) {

if(end < start) {

return false;

}

int mid = (start + end)/2;

int r = mid/col;

int c = mid%col;

// cout<< start<< " "<< mid<<" "<<end<< " "<<arr[r][c]<< " "<<element<<endl;

if(arr[r][c] == element) {

return true;

}

if(arr[r][c] < element) {

return binary\_search2d(mid+1, end, rows, col, element);

}

return binary\_search2d(start, mid - 1, rows, col, element);

}

int findMinIdx(int arr1[], int start, int end, int element, int n) {

if (start == end ){

return start;

}

int mid = (start + end)/2;

cout<<start<<" "<<end<< " "<<mid<<endl;

if((mid==0 and arr1[mid] < arr1[mid + 1]) or (mid==n-1 and arr1[mid] < arr1[mid-1]) or (arr1[mid] < arr1[mid+1]

and (arr1[mid] < arr1[mid-1]))) {

return mid;

}

if(arr1[mid] < arr1[0]) {

findMinIdx(arr1, start, mid-1, element, n);

} else {

findMinIdx(arr1, mid+1, end, element, n);

}

}

bool isElementPresent(int arr1[], int start, int end, int element) {

int idx = findMinIdx(arr1, start, end, element, end-start+1);

cout<<"MinIdx: "<< idx<<endl;

if(idx == 0){

return binary\_search(arr1, arr1+end-start+1, element);

} else {

if(element >= arr1[0]) {

return binary\_search(arr1, arr1+idx, element);

} else {

return binary\_search(arr1+idx, arr1+end-start+1, element);

}

}

}

int performInteractiveSession(int start, int end){

if(start == end){

return start;

}

int idx;

cout<<start<<" "<<end<<endl;

cin>>idx;

int mid = (start+end)/2;

if(idx > mid){

int idx1;

if(mid+1 == end) {

return performInteractiveSession(start, mid);

}

cout<<mid+1<<" "<<end<<endl;

cin>>idx1;

if(idx == idx1) {

return performInteractiveSession(mid+1, end);

} else {

return performInteractiveSession(start, mid);

}

} else {

int idx1;

cout<<start<<" "<< mid<<endl;

if(start == mid) {

return performInteractiveSession(mid+1, end);

}

cin>>idx1;

if(idx == idx1){

return performInteractiveSession(start, mid);

} else {

return performInteractiveSession(mid+1, end);

}

}

}

int main() {

int rows = 4, col = 5, total = 20;

int element = 52;

bool ok = binary\_search2d(0, total - 1, rows, col, element);

if (ok ){

cout<<"Element Found!!<"<<endl;

} else {

cout<<"Element Not Found!!"<<endl;

}

// Pivot Element

int arr1[] = {12, 13, 14, 16, 18, 19, 4, 9, 10, 11};

element = 14;

ok = isElementPresent(arr1, 0, 9, element);

if(ok) {

cout<<"Element Found!!"<<endl;

} else {

cout<<"Element Not Found!!"<<endl;

}

// Interactive 1, 7, 2, 10, 3, 11, 4

cout<<"Starting interactive session"<<endl;

int start = 0, end = 6;

int mxIdx = performInteractiveSession(start, end);

cout<<"Max Idx is: " << mxIdx;

}